Collection des rapports de

Télécom Bretagne

RR-2013-03-SC

Blind Equalization Based on pdf distance criteria and Performance Analysis

Souhaila Fki (Télécom Bretagne)
Malek Messai(Télécom Bretagne)
Abdeldjalil Aissa-El-Bey (Télécom Bretagne)
Thierry Chonavel (Télécom Bretagne)

TELECOM Bretagne

Institut Mines-Télécom

Blind Equalization Based on pdf distance criteria and Performance Analysis

Souhaila Fki*, *Graduate Student Member, IEEE,* Malek Messai, Abdeldjalil Aïssa-El-Bey, *Senior Member, IEEE,* and Thierry Chonavel, *Member, IEEE,*

Institut Télécom; Télécom Bretagne; UMR CNRS 3192 Lab-STICC

Technopôle Brest Iroise CS 83818 29238 Brest, France

Université européenne de Bretagne

Email: Firstname.Lastname@telecom-bretagne.eu

Abstract

In this report, we address M-QAM blind equalization by fitting the probability density functions (pdf) of the equalizer output with the constellation symbols. We propose two new cost functions, based on kernel pdf approximation, which force the pdf at the equalizer output to match the known constellation pdf. The kernel bandwidth of a Parzen estimator is updated during iterations to improve the convergence speed and to decrease the residual error of the algorithms. Unlike related existing techniques, the new algorithms measure the distance error between observed and assumed pdfs for the real and imaginary parts of the equalizer output separately. The advantage of proceeding this way is that the distributions show less modes, which facilitates equalizer convergence, while as for multi-modulus methods phase recovery keeps being preserved. The proposed approaches outperform CMA and classical pdf fitting methods in terms of convergence speed and residual error. We also analyse the convergence properties of the most efficient proposed equalizer via the ordinary differential equation (ODE) method.

I. Introduction

In transmissions, multipath propagation introduces intersymbols interference (ISI) that can make it difficult to recover transmitted data. Thus, an equalizer must be used to reduce the ISI. Without knowledge of the channel, the first equalization methods rely on periodic transmission of training sequences that are known from the receiver. Then, adaptation of the equalizer coefficients is done by minimizing a cost function that measures some distance between the actual equalizer output and the desired reference sequence. When the transmitter sends a training sequence, the equalizer taps can be easily adapted by using simple optimization criteria such as the Least Mean Squares (LMS) criterion that minimizes the expectation of the squared error [1]. However, in many digital communication systems, the transmission of a bandwidth consuming training sequence is not suitable. In order to avoid training, blind equalization techniques have been developped to retrieve symbols transmitted through an unknown channel by only using received data and some knowledge upon the statistics of the original sequence. There exist many blind algorithms. Sato algorithm [2] was the first blind technique proposed. The Godard algorithm [3] and the Constant Modulus Algorithm (CMA) [4] which is a particular case of Godard algorithm, are probably the most popular blind equalization techniques. However, they require a long data sequence to converge and show relatively high residual error. To overcome these limitations, several approaches have been proposed in the literature like the Modified Constant Modulus Algorithm (MCMA), that performs blind equalization and carrier phase recovery simultaneously [5], the Multi-Modulus Algorithm (MMA) that measures the errors of the real part and imaginary part of the equalizer output separately [6] and the Normalized-CMA (NCMA), that accelerates convergence by estimating the optimal step size of the algorithm at each iteration [7].

In the last decade, new blind equalization techniques, based on information theoretic criteria and pdf estimation of transmitted data, have been proposed. These criteria are optimized adaptively, in general by means of stochastic gradient techniques. Among these techniques, Renyi's entropy has been used as a cost function [8]. It involves pdf estimation with the Parzen window kernel method. This equalizer is very sensitive to noise and provides excellent results for some channels but fails to equalize some others. So, an alternative criterion based on forcing the pdf at the equalizer output to match the known constellation pdf has been proposed in [9]. As a cost function, it uses the Kullback-Leibler Divergence (KLD) between the pdfs. The Euclidean distance has also been proposed in [10]. It uses Parzen window with Gaussian kernels for pdf estimation. In [11], a technique based on fitting the pdf of the equalizer output at some relevant points that are determined by the modulus of the constellation symbols was proposed. It is known as sampled-pdf fitting. The authors of [11] also proposed in [12] the square quadratic distance (SQD) algorithm which involves fitting the whole pdf. This method is designed for multilevel modulations and works at symbol rate.

Many digital transmission systems with a high number of states use QAM modulations. As the multi-modulus approaches are well suited for such modulutions, we propose to use these techniques to equalize QAM constellations. Therefore, in this report, we propose two new blind algorithms based on the SQD fitting, that we call Multi-Modulus SQD-l2 (MSQD-l2) and MSQD-l1. Unlike the method in [12], MSQD-l2 measures the distance error between observed and assumed pdfs for the real and imaginary parts of the equalizer output separately. The advantage of proceeding this way is that involved distributions show less modes, leading thus to reduced complexity, while preserving phase recovery as for multi-modulus methods. In addition, we benefit from the fact that 1D pdfs can be accurately estimated with less data than 2D pdfs. MSQD-l1 works like MSQD-l2 but without squaring the real and the imaginary parts of the equalizer outputs. Instead, their absolute values are considered. Thus, the shape of equalized constellation modes is Gaussian which is in accordance with the statistical behavior of received data from a single path propagation, what the equalizer tries to achieve.

These techniques are designed for multilevel modulations, work at the symbol rate and admit a simple stochastic gradient-based implementation. For pdf estimation, we use the Parzen window. The proposed methods outperform CMA and classical pdf fitting approaches, in terms of convergence speed and residual error. As much as possible, it is interesting to analyze the convergence properties of blind equalizers to better understand their performance. In this report, we focus on performance anlysis of the MSQD-l1 which is the most efficient of both algorithms that we present. To this goal, we employ the Ordinary Differential Equation (ODE) method. Indeed, the ODE approach supplies a solid theoretical framework for such a task [13]. The exact convergence analysis of adaptive blind equalization algorithms is often difficult because they are derived from nonlinear criteria. Therefore, the convergence analysis of the MSQD-l1 is conducted under some usual assumptions that are commonly met in the related literature. The contributions of this work to the field of blind equalization include:

1) Two new blind equalization algorithms, MSQD-l1 and MSQD-l2, that converge faster than the CMA and the classical SQD pdf fitting [12] and achieve lower residual error;
2) Convergence and performance analysis of the most effective MSQD-l1 algorithm based on the ODE method.

This report is organized as follows. In section II, we explain why to use pdfs for blind equalization. In section III, we present the blind equalization problem and some techniques, in the literature, based on pdfs. In section IV, we propose the new cost functions and their corresponding stochastic gradient expressions. The convergence and performance analysis of the MSQD-l1 algorithm is developed in section V. Simulations are presented in section VI and conclusions of our work are given in section VII.

II. State of the art about blind equalization based on probability density functions

A. Why to use probability density functions for blind equalization channels?

Most blind equalization methods are based on statistical informations upon the transmitted symbols. The goal is to find a cost function that enables to adapt the equalizer in such a way that it produces a signal that has the same statistical properties as the emitted sequence. To extract much or ideally full statistical information from the signal of interest, the equalizer has to use high order statistics (HOS). However, a good quality of HOS estimators can not be achieved when calculated from a limited data set. That's why an order of less than or equal to fourth order statistics is usually considered. But, data distribution contains more information than the statistical properties employed in classical blind equalization methods such as the CMA that uses the fourth order statistic. Then, finding other criteria for blind equalization that better exploit statistical information about transmitted data would be interesting. Moreover, the statistics of a random variable are obtained from its distribution. Thus, it can be thought that blind equalization using probability density functions of transmitted data could achieve better performance than blind equalization based on statistical properties. Indeed, it has been shown that blind equalization methods based on probability density functions of observed and transmitted signals are more efficient than methods based on a limited set of statistical features. The idea behind blind equalization channels using pdfs is simple: knowing the target pdf of transmitted data (pdf of a constellation) in the ideal case when propagating signal is only corrupted by an additive white Gaussian noise (AWGN), we can look for an equalizer that outputs data with the same noisy constellation pdf. In the case of an AWGN channel and independent identically distributed transmitted symbols, the target pdf is a mixture of Gaussian modes that have the same probability. Fig.1 bellow summarizes the idea of blind equalization based on pdf fitting.

In the next section, we introduce the signal model and recall some blind equalization methods based on pdf fitting that can be found in the literature.

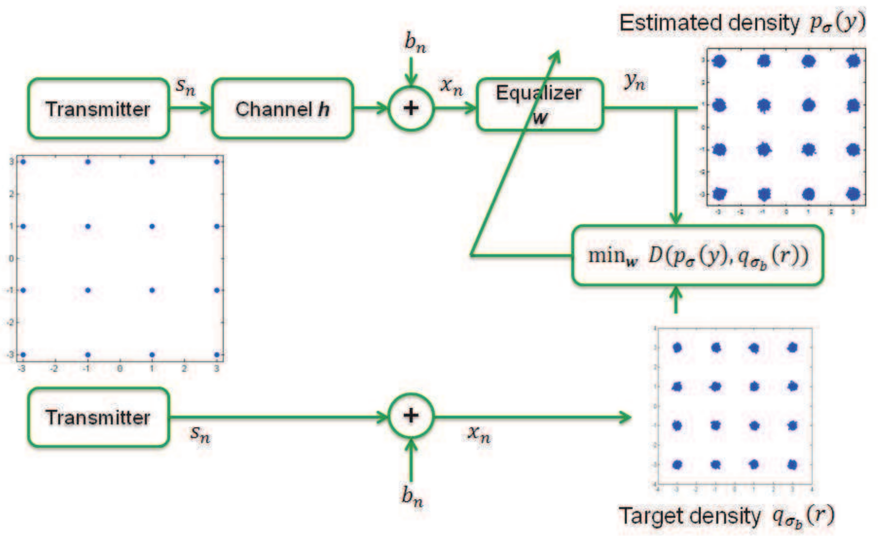

Fig. 1. The idea behind blind equalization based on pdf fitting

B. Kernel density function estimation

Let us consider any random variable with probability density function f. Kernel density estimation supplies a density function estimate $\hat{f}$ from observed data $x_1, ..., x_n$ drawn from f, that has the form bellow [14]:

$$\hat{f}(x) = \frac{1}{nh}\sum_{i=1}^{n} K(\frac{x - x_i}{h}) = \frac{1}{n}\sum_{i=1}^{n} K_h(x - x_i) \tag{1}$$

where, $K_h(t) = h^{-1}K(h^{-1}t)$. The kernel estimate is a mixture density, which has n identical component densities shifted around the data points. Any probability density may be chosen as a kernel. In this report, we use the Parzen estimator of pdfs [15] that involves a Gaussian kernel, as shown hereafter.

III. SIGNAL AND EQUALIZER MODELS

A. Signal model

To transmit digital data, a sequence $\{s(n)\}_{n\in\mathbb{Z}}$ of independent identically distributed (i.i.d) complex symbols belonging to a digital modulation constellation is sent through a propagation channel. Transmitted data are affected by multipath propagation, resulting in intersymbol interference (ISI) at the receiver side. To combat the ISI, an equalizer is used. In our work, we are interested in blind equalization, that only requires knowledge of the modulation used to send the data. The basic scheme of a blind equalization system is described in Fig.2. We assume that the symbols $\{\boldsymbol{s}(n)\}_{n\in\mathbb{Z}}$ are drawn from a symmetric QAM constellation.

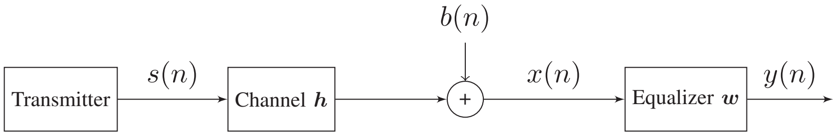

Fig. 2. Basic scheme of a blind equalization system

Fig.2 summarizes the transmission model, where the channel is modeled by its impulse response $\boldsymbol{h} = [h_0, h_1, ..., h_{L_h-1}]^T$, with $(.)^T$ denotes the transpose operator. $b = \{b(n)\}_{n\in\mathbb{Z}}$ is a circular complex additive white Gaussian noise, independent from s with variance $\sigma_b^2 = \mathbb{E}\left[|b(n)|^2\right]$, $x = \{x(n)\}_{n\in\mathbb{Z}}$ is the equalizer input, $\boldsymbol{w} = [w_0, w_1, ..., w_{L_w-1}]^T$ is the equalizer impulse response, with length L_w and $y(n)$ is the equalized signal at time n.

$x(n)$ and $y(n)$ can be modeled as

$$x(n) = \sum_{i=0}^{L_h-1} h_i\, s(n-i) + b(n) \tag{2}$$

and

$$y(n) = \sum_{i=0}^{Lw-1} w_i\, x(n-i) = \boldsymbol{w}(n)^T \boldsymbol{x}(n) \tag{3}$$

where $\boldsymbol{x}(n) = [x(n), x(n-1), ..., x(n-L_w+1)]^T$.

The weights of the equalizer will be adapted by using a stochastic gradient algorithm in the form

$$\boldsymbol{w}(n+1) = \boldsymbol{w}(n) - \mu\, \nabla_{\boldsymbol{w}} J(\boldsymbol{w}) \tag{4}$$

where μ is the step size and $J(\boldsymbol{w})$ is the cost function to be minimized.

The most popular blind algorithms are the family of Godard algorithms [3], which are stochastic gradient descent methods for minimizing the cost functions

$$J_G(\boldsymbol{w}) = \mathbb{E}[(|y(n)|^p - R_p)^2]\ p = 1, 2, ... \tag{5}$$

where, $R_p = \frac{\mathbb{E}[|s(n)|^{2p}]}{\mathbb{E}[|s(n)|^p]}$ and $\mathbb{E}[.]$ denotes mathematical expectation. For the particular case where $p = 2$, $J_G(\boldsymbol{w})$ is the cost function of the CMA [4], which was independently developed using the idea of penalizing the output samples that do not have the constant modulus property. With a stochastic gradient optimization approach, the CMA update can be written as:

$$\boldsymbol{w}(n+1) = \boldsymbol{w}(n) - \mu\,(|y(n)|^2 - R_2)y(n)x(n)^* \tag{6}$$

where, the subscript * denotes complex conjugate.

B. Blind equalization techniques based on pdf fitting in the litterature

1) Blind equalization with Renyi's entropy [16]: Instead of minimizing the squared error deviations from the desired constant modulus property as with the CMA, the authors of [16] proposed to minimize the entropy of the deviations. They considered that Shannon's entropy, the most widely known definition of entropy, was too hard to estimate and minimize and proposed alternatively to minimize the order-α Renyi's entropy which for a random variable with pdf f, is defined as [17]

$$H_\alpha(f) = \frac{1}{1-\alpha} log\Big(\int_{-\infty}^{+\infty} f(x)^\alpha dx\Big) \tag{7}$$

The following family of entropy-based cost functions was used in [16]:

$$J_\alpha^p(\boldsymbol{w}) = H_\alpha(y(n)^p - R_p)\ p = 1, 2, ... \tag{8}$$

Since the entropy does not depend on the mean of the signal, the following simple criteria were used equivalently:

$$J_\alpha^p(\boldsymbol{w}) = H_\alpha(y(n)^p)\ p = 1, 2, ... \tag{9}$$

The authors of [16] focussed on the case $p = 2$. It was shown in [16] that the minimization of the entropy cost function Eq.(9) is equivalent, for $\alpha > 1$, to the maximization of $V_\alpha(\boldsymbol{w}) = \mathbb{E}[f(|y(n)|^2)^{\alpha-1}]$. Using a window of N samples formed by the current and the past $N-1$ outputs of the equalizer, the information potential can be estimated by substituting the expectation by a sample mean:

$$V_\alpha(\boldsymbol{w}) \approx \frac{1}{N}\sum_j f(|y(j)|^2)^{\alpha-1} \tag{10}$$

Then, the coefficients of the equalizer are updated by

$$\boldsymbol{w}(n+1) = \boldsymbol{w}(n) + \mu\frac{\partial V_\alpha(\boldsymbol{w})}{\partial \boldsymbol{w}} \tag{11}$$

where μ is the stepsize of the algorithm. The authors of [16] showed that the use of quadratic entropy ($\alpha = 2$) and a short window ($N = 2$) is recommended since in this case an algorithm with a computational cost similar to that of the CMA, but with a much faster convergence, is obtained.

2) Blind equalization with the Kullback-Leibler Divergence (KLD) [18]: The idea behind this technique is to compare the KLD between the observed pdf and the target one. With an ideal linear equalizer, the output of the equalizer can be written as

$$\begin{aligned} y(n) &= (\boldsymbol{H}\boldsymbol{s}(n) + \boldsymbol{b}(n))^T \boldsymbol{w}_{\text{ideal}} \\ &= \boldsymbol{s}^T(n)\boldsymbol{g}_{\text{ideal}} + \mathrm{b}(n) \\ &= s(n-\delta) + \mathrm{b}(n) \end{aligned} \tag{12}$$

where, $\boldsymbol{g}_{\text{ideal}}$ is the ideal system response, δ is the system delay thus $s(n-\delta)$ is the transmitted symbol at time $n-\delta$ and $\boldsymbol{b}(n)$ is a random variable assumed to be Gaussian. From the last equation Eq.(12), it is clear that the pdf of the signal at the optimal equalizer output is a mixture of equiprobable Gaussian distributions with the same variance $\sigma_b^2 = \mathbb{E}\left[|b(n)|^2\right]$. Then, the target pdf is also a mixture of equiprobable Gaussians:

$$p_{Y,\text{ideal}}(y) = \frac{1}{N_s\sqrt{2\pi\sigma_{\boldsymbol{b}}^2}} \sum_{i=1}^{N_s} e^{\frac{-|y(n)-s(i)|^2}{2\sigma_{\boldsymbol{b}}^2}} \tag{13}$$

where, N_s is the number of complex symbols in the constellation. Thus, the equalization criterion is constructed in such a way that it forces the adaptive filter to produce signals with the same pdf than the ideal one. In order to use the KLD, the authors of [18] constructed a parametric model of Gaussian mixture, like that one in Eq.(13). Thus, the observed pdf was estimated by the following parametric model as it was explained in [18]:

$$\phi(y,\sigma_r^2) = \frac{1}{N_s\sqrt{2\pi\sigma_r^2}} \sum_{i=1}^{N_s} e^{-\frac{|y(n)-a_i|^2}{2\sigma_r^2}} \tag{14}$$

where, σ_r^2 is the variance of each Gaussian in the model which is different from σ_b^2 until the optimal equalizer is not yet reached. Then, the KLD is expressed by:

$$D_{p_{Y,\text{ideal}}(y)||\phi(y,\sigma_r^2)} = \int_{-\infty}^{\infty} p_{Y,\text{ideal}}(y) ln\Big(\frac{p_{Y,\text{ideal}}(y)}{\phi(y,\sigma_r^2)}\Big) dy \tag{15}$$

For simplicity, instead of minimizing the KLD, the authors of [18] minimized only the $\phi(y,\sigma_r^2)$ dependent term. They considered then the following cost function:

$$\begin{aligned} J_{FP}(\boldsymbol{w}) &= -\mathbb{E}\left\{ln[\phi(y,\sigma_r^2)]\right\} \\ &= -\mathbb{E}\left\{ln\Big[\frac{1}{N_s\sqrt{2\pi\sigma_r^2}} \sum_{i=1}^{N_s} e^{-\frac{|y(n)-a_i|^2}{2\sigma_r^2}}\Big]\right\} \end{aligned} \tag{16}$$

and, the equalizer was adapted by

$$\boldsymbol{w}(n+1) = \boldsymbol{w}(n) - \mu \nabla_{\boldsymbol{w}} J_{FP}(\boldsymbol{w}) \tag{17}$$

3) SQD pdf fitting using Parzen Estimator [12]: As it was mentioned in the last section when using the KLD criterion, the first term of the KLD was dropped from the cost function, since it can not be easily evaluated or estimated. By dropping it, important statistical information about the pdf of the actual random variable Y is eliminated. To avoid this drawback, a stochastic algorithm based on pdf fitting was proposed. Equalization techniques based on pdf matching intend to minimize some distance between the data distribution at the equalizer output and some target distribution. Transmitted symbols have a discrete distribution. But, since they are affected by additive Gaussian noise at the receiver side, it can be assumed that after removing channel multipath effects, the equalizer output should consist of a Gaussian mixture, with Gaussian modes centered at the constellation points. So, a target distribution of this form can be chosen. In [12], a quadratic distance between pdfs for the cost function was proposed. It is given by

$$J(\boldsymbol{w}) = \int_{-\infty}^{\infty} \big(f_{Y^p}(z) - f_{S^p}(z)\big)^2 dz \tag{18}$$

where, $Y^p = \{|y(n)|^p\}$, $S^p = \{|s(n)|^p\}$ and $f_Z(z)$ denotes the pdf of Z at z. Thus, $J(\boldsymbol{w})$ is intended to match p^{th} moment distributions between the equalizer output and the noisy constellation.

The Parzen window method is used to estimate the current data pdf. Using this nonparametric pdf estimator with the L last symbols, the estimates of the pdfs at time n are given by:

$$\begin{aligned} \hat{f}_{Y^p}(z) &= \frac{1}{L}\sum_{k=0}^{L-1} K_{\sigma_0}(z - |y(n-k)|^p) \\ \hat{f}_{S^p}(z) &= \frac{1}{N_s}\sum_{k=1}^{N_s} K_{\sigma_0}(z - |s(k)|^p) \end{aligned} \tag{19}$$

where N_s is the number of complex symbols in the constellation and K_{σ_0} is a Gaussian kernel with standard deviation σ_0, also known as the kernel bandwidth:

$$K_{\sigma_0}(x) = \frac{1}{\sqrt{2\pi}\sigma_0} e^{-\frac{x^2}{2\sigma_0^2}}. \tag{20}$$

According to [12], for $p = 2$ and $L = 1$, the expression of the cost function is given by

$$J(\boldsymbol{w}) = \frac{1}{N_s^2}\sum_{k=1}^{N_s}\sum_{l=1}^{N_s} K_{\sigma}(|s(l)|^2 - |s(k)|^2) - \frac{2}{N_s}\sum_{k=1}^{N_s} K_{\sigma}(|y(n)|^2 - |s(k)|^2). \tag{21}$$

where, $\sigma = \sqrt{2}\sigma_0$. Then, the gradient of $J(\boldsymbol{w})$ with respect to the equalizer weights is given by

$$\nabla_{\boldsymbol{w}} J(\boldsymbol{w}) = -\frac{1}{N_s} \sum_{k=1}^{N_s} K'_{\sigma}(|y(n)|^2 - |s(k)|^2)\, y(n)\, \boldsymbol{x}^*(n) \tag{22}$$

where $K'_{\sigma}(x) = -\frac{x}{\sqrt{2\pi}\sigma^3} \exp(\frac{-x^2}{2\sigma^2})$ is the derivative of $K_\sigma(x)$ and $(.)^*$ denotes the complex conjugation operator. Then, the equalizer coefficients are updated at symbol rate by inserting Eq.(22) in Eq.(4). This algorithm is initialized with a tap-centered equalizer. In [12], the squared modulus of the symbols for the kernel variables $(p = 2)$ is used to design $J(\boldsymbol{w})$. But, squaring does not preserve Gaussianity around noisy constellation points. With a view to make the criterion statistically more meaningful we propose, in this report, to also address the case $p = 1$. Indeed, when $p = 1$, since constellation points are apart from the axes, at convergence $|y(n)|$ will be roughly distributed according to a mixture of gaussian distributions around the constellation points that are in the positive quadrant of the complex plane. This is true provided the SNR remains in usual ranges for QAM modulations under consideration. In addition, it is well known that multimodulus approaches such as MMA [19], that decompose equalization criteria into an in-phase term and a quadrature one, are more efficient than criteria such as the CMA [4], that handle in-phase and quadrature parts together. In the same way, the criteria that we propose are made of a sum of two terms related to in-phase and quadrature parts of the equalizer output. This will lead to criteria that we name Multimodulus-SQD (MSQD). More specifically, for $p = 1$ and $p = 2$ we get MSQD-l1 and MSQD-l2.

IV. MSQD ALGORITHMS

A. MSQD-ℓp algorithm

MSQD family consists of algorithms based on cost functions in the form:

$$J(\boldsymbol{w}) = \int_{-\infty}^{\infty} (\hat{f}_{|y_r|^p}(z) - \hat{f}_{|s_r|^p}(z))^2 dz + \int_{-\infty}^{\infty} (\hat{f}_{|y_i|^p}(z) - \hat{f}_{|s_i|^p}(z))^2 dz \tag{23}$$

where $y_r = \Re\{y\}$, $y_i = \Im\{y\}$ and the pdf estimates are in the form

$$\hat{f}_x(z) = \frac{1}{N_x} \sum_{k=1}^{N_x} K_{\sigma_0}(z - x_k) \tag{24}$$

x is equal to $|s_r|^p$, $|s_i|^p$, $|y_r|^p$ or $|y_i|^p$. $N_x = N_s$ for $x = |s_{r,i}|^p$ and $N_x = L$ for $x = |y_{r,i}|^p$.

For fixed p, we denote the corresponding criterion by MSQD-ℓp. Expending Eq.(23), we get

$$\begin{aligned} J(\boldsymbol{w}) &= \int_{-\infty}^{\infty} \hat{f}_{|y_r|^p}(z)^2\, dz - 2\int_{-\infty}^{\infty} \hat{f}_{|y_r|^p}(z)\, \hat{f}_{|s_r|^p}(z)\, dz + \int_{-\infty}^{\infty} \hat{f}_{|s_r|^p}(z)^2\, dz + \int_{-\infty}^{\infty} \hat{f}_{|y_i|^p}(z)^2 dz \\ &- 2\int_{-\infty}^{\infty} \hat{f}_{|y_i|^p}(z)\hat{f}_{|s_i|^p}(z)\, dz + \int_{-\infty}^{\infty} \hat{f}_{|s_i|^p}(z)^2\, dz \end{aligned} \tag{25}$$

Then, according to Eq.(24), we obtain the following expression for $J(\boldsymbol{w})$

$$\begin{aligned} J(\boldsymbol{w}) &= \frac{1}{L^2}\sum_{k=0}^{L-1}\sum_{l=0}^{L-1}\int_{-\infty}^{\infty} K_{\sigma_0}(z-|y_r(n-k)|^p)K_{\sigma_0}(z-|y_r(n-l)|^p)\, dz \\ &- \frac{2}{N_s L}\sum_{k=1}^{N_s}\sum_{l=0}^{L-1}\int_{-\infty}^{\infty} K_{\sigma_0}(z-|y_r(n-l)|^p)K_{\sigma_0}(z-|s_r(k)|^p)\, dz \\ &+ \frac{1}{N_s^2}\sum_{k=1}^{N_s}\sum_{l=1}^{N_s}\int_{-\infty}^{\infty} K_{\sigma_0}(z-|s_r(k)|^p)K_{\sigma_0}(z-|s_r(l)|^p)\, dz \\ &+ \frac{1}{L^2}\sum_{k=0}^{L-1}\sum_{l=0}^{L-1}\int_{-\infty}^{\infty} K_{\sigma_0}(z-|y_i(n-k)|^p)K_{\sigma_0}(z-|y_i(n-l)|^p)\, dz \\ &- \frac{2}{N_s L}\sum_{k=1}^{N_s}\sum_{l=0}^{L-1}\int_{-\infty}^{\infty} K_{\sigma_0}(z-|y_i(n-l)|^p)K_{\sigma_0}(z-|s_i(k)|^p)\, dz \\ &+ \frac{1}{N_s^2}\sum_{k=1}^{N_s}\sum_{l=1}^{N_s}\int_{-\infty}^{\infty} K_{\sigma_0}(z-|s_i(k)|^p)K_{\sigma_0}(z-|s_i(l)|^p)\, dz \end{aligned} \tag{26}$$

In a stochastic gradient optimization approach, in general only instantaneous statistics are involved in the criterion. Thus, we consider a window length $L = 1$ as in [12]. Then, since for Gaussian kernels we have

$$\int_{-\infty}^{\infty} K_{\sigma_0}(y-C_1)K_{\sigma_0}(y-C_2)dy = \frac{1}{2}K_{\sigma_0\sqrt{2}}(C_1-C_2), \tag{27}$$

Thus, $J(\boldsymbol{w})$ becomes

$$J(\boldsymbol{w}) = -\frac{1}{N_s}\sum_{k=1}^{N_s} K_\sigma(|y_r(n)|^p - |s_r(k)|^p) - \frac{1}{N_s}\sum_{k=1}^{N_s} K_\sigma(|y_i(n)|^p - |s_i(k)|^p) + Cst. \tag{28}$$

On another hand, $y(n) = \boldsymbol{w}(n)^T\boldsymbol{x}(n)$ rewrites as

$$y(n) = [\boldsymbol{w}_r^T\boldsymbol{x}_r(n) - \boldsymbol{w}_i^T\boldsymbol{x}_i(n)] + j[\boldsymbol{w}_r^T\boldsymbol{x}_i(n) + \boldsymbol{w}_i^T\boldsymbol{x}_r(n)], \tag{29}$$

which leads to $\frac{\partial y(n)}{\partial \boldsymbol{w}_r} = \boldsymbol{x}(n)$ and $\frac{\partial y(n)}{\partial \boldsymbol{w}_i} = j\,\boldsymbol{x}(n)$.

Therefore, the derivative of $J(\boldsymbol{w})$ with respect to equalizer weights is

$$\begin{aligned}
\nabla_{\boldsymbol{w}} J(\boldsymbol{w}) &= \frac{\partial J(\boldsymbol{w})}{\partial \boldsymbol{w}_r} + j\,\frac{\partial J(\boldsymbol{w})}{\partial \boldsymbol{w}_i} \\
&= \frac{1}{2}\left[\frac{\partial J(\boldsymbol{w})}{\partial y_r(n)}\frac{\partial y_r(n)}{\partial \boldsymbol{w}_r} + \frac{\partial J(\boldsymbol{w})}{\partial y_i(n)}\frac{\partial y_i(n)}{\partial \boldsymbol{w}_r} + j\,\left(\frac{\partial J(\boldsymbol{w})}{\partial y_r(n)}\frac{\partial y_r(n)}{\partial \boldsymbol{w}_i} + \frac{\partial J(\boldsymbol{w})}{\partial y_i(n)}\frac{\partial y_i(n)}{\partial \boldsymbol{w}_i}\right)\right] \\
&= \frac{1}{2}\left[\frac{\partial J}{\partial y_r(n)}\boldsymbol{x}_r(n) + \frac{\partial J}{\partial y_i(n)}\boldsymbol{x}_i(n) + j\,\left(\frac{-\partial J}{\partial y_r(n)}\boldsymbol{x}_i(n) + \frac{\partial J(\boldsymbol{w})}{\partial y_i(n)}\boldsymbol{x}_r(n)\right)\right] \\
&= \frac{1}{2}\left[\left(\frac{\partial J}{\partial y_r(n)} + j\,\frac{\partial J}{\partial y_i(n)}\right)\boldsymbol{x}^*(n)\right] \\
&= \frac{p}{2\sqrt{2\pi}N_s\sigma^3}\sum_{k=1}^{N_s}\Big(\text{sign}\,(y_r(n))\,|y_r(n)|^{p-1}\,(|y_r(n)|^p - |s_r(k)|^p)\;e^{-\frac{(|y_r(n)|^p - |s_r(k)|^p)^2}{2\sigma^2}} \\
&\quad + j\,\text{sign}\,(y_i(n))\,|y_i(n)|^{p-1}\,(|y_i(n)|^p - |s_i(k)|^p)\;e^{-\frac{(|y_i(n)|^p - |s_i(k)|^p)^2}{2\sigma^2}}\Big)\boldsymbol{x}^*(n).
\end{aligned} \tag{30}$$

B. MSQD-ℓ2 and MSQD-ℓ1 algorithms

In the following, we focus on the cases $p = 2$ and $p = 1$.

For $p = 2$, we get from Eq.(30)

$$\begin{aligned}
\nabla_{\boldsymbol{w}} J(\boldsymbol{w}) &= \frac{1}{\sqrt{2\pi}N_s\sigma^3}\sum_{k=1}^{N_s}\Big(y_r(n)(|y_r(n)|^2 - |s_r(k)|^2)\,e^{-\frac{(|y_r(n)|^2 - |s_r(k)|^2)^2}{2\sigma^2}} \\
&+ j\,y_i(n)(|y_i(n)|^2 - |s_i(k)|^2)\,e^{-\frac{(|y_i(n)|^2 - |s_i(k)|^2)^2}{2\sigma^2}}\Big)\boldsymbol{x}^*(n)
\end{aligned} \tag{31}$$

Then, Eq.(4) is used to update equalizer taps.

For $p = 1$ and for the MSQD-ℓ1 cost function, we get an updating term of the equalizer in the form:

$$\begin{aligned}
\nabla_{\boldsymbol{w}} J(\boldsymbol{w}) &= \frac{1}{2\sqrt{2\pi}N_s\sigma^3}\sum_{k=1}^{N_s}\Big(\text{sign}(y_r(n))\,(|y_r(n)| - |s_r(k)|)\,e^{-\frac{(|y_r(n)| - |s_r(k)|)^2}{2\sigma^2}} \\
&+ j\,\text{sign}(y_i(n))\,(|y_i(n)| - |s_i(k)|)\,e^{-\frac{(|y_i(n)| - |s_i(k)|)^2}{2\sigma^2}}\Big)\boldsymbol{x}^*(n) \\
&= \phi(y(n))\,\boldsymbol{x}^*(n)
\end{aligned} \tag{32}$$

In section VI, we will show on simulations that, as expected from the discussion at the end of section III, the MSQD-ℓ1 algorithm that we propose is more effective than the existing SQD algorithm in terms of mean square error, especially for larger constellations. Thus, in the following section, we will restrict our interest to performance analysis of the MSQD-ℓ1 algorithm in terms of stationary stable points and asymptotic steady state.

V. Performance Analysis

In order to analyze asymptotic performance of the MSQD-$\ell 1$ algorithm, we resort to the ODE method which is a powerful tool for studying the behavior of stochastic algorithms of the form

$$\boldsymbol{\theta}(n) = \boldsymbol{\theta}(n-1) + \mu_n \, H\big(\boldsymbol{\theta}(n-1), \boldsymbol{x}(n)\big) \tag{33}$$

where $\{\boldsymbol{x}(n)\}_{n\in\mathbb{N}}$ is assumed to be a Markov and ergodic process. So, we can assume that there exists a function $h(\boldsymbol{\theta})$ such that

$$h(\boldsymbol{\theta}) = \lim_{n\to+\infty} \mathbb{E}\big[H(\boldsymbol{\theta}, \boldsymbol{x}_n)\,|\boldsymbol{\theta}\big] \tag{34}$$

Then, the ODE is defined by the following equation

$$\frac{d\boldsymbol{\theta}}{dt} = h(\boldsymbol{\theta}). \tag{35}$$

We are going to look for stationary points of Eq.(35), that satisfy $h(\boldsymbol{\theta}) = 0$.

A. Stationary Stable Points of the ODE

In this section, we denote by $\boldsymbol{\theta}(t)$ the solution of the ODE. It depends on the initial value $\boldsymbol{\theta}(0) = \boldsymbol{\theta}_0$. A point $\boldsymbol{\theta}_*$ is a stationary point of the ODE if $h(\boldsymbol{\theta}_*) = 0$ and a stationnary point is said to be

- stable if $\forall\, \epsilon > 0, \exists\, \nu > 0,$ such as $|\boldsymbol{\theta}_0 - \boldsymbol{\theta}_*| < \nu \Rightarrow \forall t \in \mathbb{R}_+, |\boldsymbol{\theta}(t) - \boldsymbol{\theta}_*| < \epsilon$;
- asymptotically stable if $\exists\, \nu > 0,$ such as $|\boldsymbol{\theta}_0 - \boldsymbol{\theta}_*| < \nu \Rightarrow \lim_{t\to+\infty} \boldsymbol{\theta}(t) = \boldsymbol{\theta}_*$.

1) Stationary points: Let us consider the stochastic MSQD-l1 algorithm:

$$\boldsymbol{w}(n+1) = \boldsymbol{w}(n) - \mu \nabla_{\boldsymbol{w}} J\big(\boldsymbol{w}(n), \boldsymbol{x}(n)\big). \tag{36}$$

Clearly, we have here $\boldsymbol{\theta}(n) = \boldsymbol{w}(n)$ and

$$H\big(\boldsymbol{\theta}(n), \boldsymbol{x}(n+1)\big) = -\nabla_{\boldsymbol{w}} J\big(\boldsymbol{w}(n), \boldsymbol{x}(n)\big) = -\nabla_{\boldsymbol{w}} J\big(\boldsymbol{w}, y(n)\big) = -\phi\big(y(n)\big)\, \boldsymbol{x}^*(n). \tag{37}$$

Then, according to Eq.(34)

$$\begin{aligned} h(\boldsymbol{w}) &= \lim_{n\to+\infty} \mathbb{E}\big[H(\boldsymbol{w}, \boldsymbol{x}(n))|\boldsymbol{w}\big] \\ &= \lim_{n\to+\infty} -\mathbb{E}\big[\nabla_{\boldsymbol{w}} J(\boldsymbol{w}, y(n))|\boldsymbol{w}\big] \\ &= -\int_{\mathbb{R}^+} \nabla_{\boldsymbol{w}} J(\boldsymbol{w}, y) p_{|Y|}(y)\, dy \end{aligned} \tag{38}$$

where, $p_{|Y|}(y)$ is the probability density of $|y|$. To calculate $h(\boldsymbol{w})$, we begin by calculating $F_{|Y|\,|s(k)}(y)$ where, $|y|\ |s(k) = |s(k)+\varepsilon|$ and $F_{|Y|\ |s(k)}(y)$ is the cumulative distribution of $|Y|$ given $s(k)$:

$$\begin{aligned} F_{(|Y|\,|s(k))}(y) &= P\Big(|Y| \leq y\ |s(k)\Big) \\ &= P\Big(-y \leq Y \leq y\ |s(k)\Big) \\ &= P\Big(\frac{-y-s(k)}{\sigma_\varepsilon} \leq \frac{Y-s(k)}{\sigma_\varepsilon} \leq \frac{y-s(k)}{\sigma_\varepsilon}\ |s(k)\Big) \\ &= F\Big(\frac{y-s(k)}{\sigma_\varepsilon}\Big) - F\Big(\frac{-y-s(k)}{\sigma_\varepsilon}\Big) \end{aligned} \tag{39}$$

where F is the cumulative distribution function of the $\mathcal{N}(0,1)$ distribution. Indeed, the ISI at the output of the equalizer can be modelled as a Gaussian distribution [20] and thus $y \sim \mathcal{N}(s(k), \sigma_\varepsilon^2)$ where σ_ε^2 is the variance of the error (ε) between y and $s(k)$. Then, from Eq.(39),

$$\begin{aligned} p_{(|Y|\,|s(k))}(y) &= \frac{1}{\sigma_\varepsilon}\Big[\mathcal{N}\Big(\frac{y-s(k)}{\sigma_\varepsilon};0,1\Big) + \mathcal{N}\Big(\frac{y+s(k)}{\sigma_\varepsilon};0,1\Big)\Big] \\ &= \mathcal{N}\Big(y-s(k);0,\sigma_\varepsilon^2\Big) + \mathcal{N}\Big(y+s(k);0,\sigma_\varepsilon^2\Big). \end{aligned} \tag{40}$$

where, $\mathcal{N}(y;m,\sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-(y-m)^2}{2\sigma^2}}$. Summing over all possible symbols s_k, we find the following expression of $p_{|Y|}(y)$:

$$p_{|Y|}(y) = \frac{1}{N_s}\sum_{k=0}^{N_s-1}\big[\mathcal{N}(y-s(k);0,\sigma_\varepsilon^2) + \mathcal{N}(y+s(k);0,\sigma_\varepsilon^2)\big]. \tag{41}$$

Then, we obtain the following expression of $h(\boldsymbol{w})$ after replacing $J(\boldsymbol{w}, y)$ by its expression and accounting for symmetry properties of $J(\boldsymbol{w}, y)$ and $p_{|Y|}(y)$,

$$h(\boldsymbol{w}) = \frac{1}{\pi N_s^2 \sigma\sigma_\varepsilon}\big[\nabla_{\boldsymbol{w}} \int_{\mathbb{R}^+} \sum_{k=1}^{N_s} e^{-\frac{(y_r-|s_r(k)|)^2}{2\sigma^2}} \sum_{l=1}^{N_s}\big(e^{-\frac{(y_r-|s_r(l)|)^2}{2\sigma_\varepsilon^2}} + e^{-\frac{(y_r+|s_r(l)|)^2}{2\sigma_\varepsilon^2}}\big) dy_r. \tag{42}$$

Thus, after calculating the integral,

$$\begin{aligned} h(\boldsymbol{w}) &= \frac{1}{N_s^2\sqrt{2\pi(\sigma^2+\sigma_\varepsilon^2)}}\Big[\nabla_{\boldsymbol{w}} \sum_{k=1}^{N_s}\sum_{l=1}^{N_s} e^{-\frac{(|s_r(k)|-|s_r(l)|)^2}{2(\sigma^2+\sigma_\varepsilon^2)}}\Big(1 - \text{erfc}\big(\frac{-|s_r(k)|\sigma_\varepsilon^2 - |s_r(l)|\sigma^2}{\sigma\sigma_\varepsilon\sqrt{\sigma_\varepsilon^2+\sigma^2}\sqrt{2}}\big)\Big) \\ &+ e^{-\frac{(|s_r(k)|+|s_r(l)|)^2}{2(\sigma^2+\sigma_\varepsilon^2)}}\Big(1 - \text{erfc}\big(\frac{-|s_r(k)|\sigma_\varepsilon^2 + |s_r(l)|\sigma^2}{\sigma\sigma_\varepsilon\sqrt{\sigma_\varepsilon^2+\sigma^2}\sqrt{2}}\big)\Big)\Big] \\ &= \nabla_{\boldsymbol{w}} A(\sigma_\varepsilon^2) \\ &= \frac{dA(\sigma_\varepsilon^2)}{d\sigma_\varepsilon^2}\nabla_{\boldsymbol{w}}\sigma_\varepsilon^2. \end{aligned} \tag{43}$$

The stationary points $\boldsymbol{w}_*$ are solutions of $h(\boldsymbol{w}_*) = 0$.

The equalized symbol $y(n)$ can be expressed by $y(n) = \boldsymbol{w}^T\widetilde{\boldsymbol{H}}\boldsymbol{s}(n) + \boldsymbol{w}^T\boldsymbol{b}(n)$ where,

$$\widetilde{\boldsymbol{H}} = \begin{pmatrix} h_0 & h_1 & \cdots & h_{L-1} & 0 & \cdots & & 0 \\ 0 & h_0 & h_1 & \cdots & h_{L-1} & 0 & \cdots & 0 \\ 0 & \ddots & h_0 & h_1 & \cdots & h_{L-1} & \vdots & \\ 0 & \vdots & & h_0 & h_1 & \cdots & h_{L-1} & 0 \\ 0 & & \cdots & 0 & h_0 & h_1 & \cdots & h_{L-1} \end{pmatrix} \tag{44}$$

and $\boldsymbol{s}(n) = \big[s(D-1), ..., s(k_n), ..., s(D-L-L_w+1)\big]^T$ where D is the equalizer delay. Then, the error between $y(n)$ and the value of the transmitted symbol $s(k_n)$ received at time n is calculated as follows

$$\begin{aligned} \sigma_\varepsilon^2 &= \mathbb{E}\Big[\Big(y(n) - s(k_n)\Big)\Big(y(n) - s(k_n)\Big)^H\Big] \\ &= \mathbb{E}\Big[\Big((\boldsymbol{w}^T\widetilde{\boldsymbol{H}} - \boldsymbol{e}_D)\boldsymbol{s}(n) + \boldsymbol{w}^T\boldsymbol{b}(n)\Big)\Big((\boldsymbol{w}^T\widetilde{\boldsymbol{H}} - \boldsymbol{e}_D)\boldsymbol{s}(n) + \boldsymbol{w}^T\boldsymbol{b}(n)\Big)^H\Big] \end{aligned} \tag{45}$$

where, $\boldsymbol{e}_D = (0, ..., 1, ..., 0)^T$, $e_D(i) = \delta_{i,D}$. Then,

$$\sigma_\varepsilon^2 = \sigma_s^2\boldsymbol{w}^T\widetilde{\boldsymbol{H}}\widetilde{\boldsymbol{H}}^H\boldsymbol{w}^* - \sigma_s^2\boldsymbol{w}^T\widetilde{\boldsymbol{H}}\boldsymbol{e}_D - \sigma_s^2\boldsymbol{e}_D{}^T\widetilde{\boldsymbol{H}}^H\boldsymbol{w}^* + \sigma_s^2 + \sigma_{\boldsymbol{b}}^2||\boldsymbol{w}||^2 \tag{46}$$

Thus,

$$h(\boldsymbol{w}) = 2\frac{dA(\sigma_\varepsilon^2)}{d\sigma_\varepsilon^2}\Big[\boldsymbol{w}^T[\sigma_s^2\widetilde{\boldsymbol{H}}\widetilde{\boldsymbol{H}}^H + \sigma_{\boldsymbol{b}}^2\boldsymbol{I}_{L_w}] - \sigma_s^2\boldsymbol{e}_D{}^T\widetilde{\boldsymbol{H}}^H\Big]^T$$

But, since in practice $|s_r(k_n)| >> \sigma^2 + \sigma_\varepsilon^2$, only terms with 0 in the exponential will be non negligible, resulting in $A(\sigma_\varepsilon^2) \approx \frac{1}{N_s^2\sqrt{2\pi(\sigma^2+\sigma_\varepsilon^2)}}\sum_{k=1}^{N_s}\big(1 + \text{erfc}(\frac{|s_r(k_n)|}{\sqrt{2}}\sqrt{\frac{1}{\sigma^2} + \frac{1}{\sigma_\varepsilon^2}})\big)$. Since at convergence, $\frac{1}{\sigma^2} >> 1$ and $\frac{1}{\sigma_\varepsilon^2} >> 1$, it is easy to check that $\frac{dA(\sigma_\varepsilon^2)}{d\sigma_\varepsilon^2} < 0$. Thereby, $\boldsymbol{w}_*^T = \sigma_s^2\boldsymbol{e}_D{}^T\widetilde{\boldsymbol{H}}^H\big[\sigma_s^2\widetilde{\boldsymbol{H}}\widetilde{\boldsymbol{H}}^H + \sigma_{\boldsymbol{b}}^2\boldsymbol{I}_{L_w}\big]^{-1}$ is the only stationnary point. The very nice thing is that $\boldsymbol{w}_*$ is the MMSE filter. Thus, we have proved that the MSQD-$\ell 1$ algorithm has only one stationnary point, which is the MMSE filter, when σ^2 and σ_ε^2 are much smaller than constellation point amplitudes.

2) Stability analysis of stationary points: Let us recall that if $\boldsymbol{w}_*$ is a stationary point of the ODE and $\lambda_1, \lambda_2, ...\lambda_{L_w}$ are the eigenvalues of $\frac{dh(\boldsymbol{w})}{d\boldsymbol{w}}|_{\boldsymbol{w}=\boldsymbol{w}_*}$, the stability of $\boldsymbol{w}_*$ is determined by the eigenvalues of $\frac{dh(\boldsymbol{w})}{d\boldsymbol{w}}|_{\boldsymbol{w}=\boldsymbol{w}_*}$. Indeed we have the following theorem [21]:

- If $\forall i \in [|1; L_w|], \Re(\lambda_i) < 0$, $\boldsymbol{w}_*$ is asymptotically stable
- If $\exists i \in [|1; L_w|], \Re(\lambda_i) > 0$, $\boldsymbol{w}_*$ is unstable
- If $\forall i \in [|1; L_w|], \Re(\lambda_i) \leq 0$ and $\Re(\lambda_{i_0}) = 0$ for $i_0 \in [|1; L_w|]$, we can not conclude from these values.

So we start by calculating $\frac{dh(\boldsymbol{w})}{d\boldsymbol{w}}|_{\boldsymbol{w}=\boldsymbol{w}_*}$.

$$\begin{aligned} \frac{dh(\boldsymbol{w})}{d\boldsymbol{w}} &= \frac{d}{d\boldsymbol{w}}\left[\frac{d}{d\sigma_\varepsilon^2}A(\sigma_\varepsilon^2)\nabla_{\boldsymbol{w}}\sigma_\varepsilon^2\right] \\ &= \frac{d^2A(\sigma_\varepsilon^2)}{d(\sigma_\varepsilon^2)^2}(\nabla_{\boldsymbol{w}}\sigma_\varepsilon^2)(\nabla_{\boldsymbol{w}}\sigma_\varepsilon^2)^T + \frac{dA(\sigma_\varepsilon^2)}{d\sigma_\varepsilon^2}\frac{d(\nabla_{\boldsymbol{w}}\sigma_\varepsilon^2)}{d\boldsymbol{w}}. \end{aligned} \tag{47}$$

Here, we are interested in the stability of the stationary point $\boldsymbol{w}_* = \sigma_s^2\left[\sigma_s^2\widetilde{\boldsymbol{H}}^*\widetilde{\boldsymbol{H}}^T + \sigma_{\boldsymbol{b}}^2\boldsymbol{I}_{L_w}\right]^{-1}\widetilde{\boldsymbol{H}}^*\boldsymbol{e}_D$. Thus,

$$\frac{dh(\boldsymbol{w})}{d\boldsymbol{w}}|_{\boldsymbol{w}=\boldsymbol{w}_*} = \frac{dA(\sigma_\varepsilon^2)}{d\sigma_\varepsilon^2}\left[\sigma_s^2\widetilde{\boldsymbol{H}}\widetilde{\boldsymbol{H}}^H + \sigma_{\boldsymbol{b}}^2\boldsymbol{I}_{L_w}\right]^T < 0 \tag{48}$$

since $(\sigma_s^2\widetilde{\boldsymbol{H}}\widetilde{\boldsymbol{H}}^H + \sigma_{\boldsymbol{b}}^2\boldsymbol{I})$ is a positive definite matrix and $\frac{dA(\sigma_\varepsilon^2)}{d\sigma_\varepsilon^2} < 0$. Thus, we have proved that the MMSE equalizer is the only stationary stable point of the MSQD-$\ell 1$ algorithm.

B. Asymptotic Steady-State MSE Analysis

1) convergence in mean: The ODE analysis holds for small step size μ. In this section, we study how it should be selected to guarantee convergence. In practice, μ should be shosen small enough. The maximum possible range for μ depends on the channel under consideration and its calculation is supplied in the Appendix (VIII-A).

2) MSE equalizer analysis: The asymptotic covariance matrix of the residual error $\boldsymbol{\epsilon}(n) = (\boldsymbol{w}_n - \boldsymbol{w}_*)$ is denoted by $\boldsymbol{\Sigma}(n) = \mathbb{E}[\boldsymbol{\epsilon}(n)\boldsymbol{\epsilon}(n)^H]$. For small step size μ, we have $\bar{y}(n+D) \approx s(k_n)$ and according to [13] (p.102 p.103) $\boldsymbol{\Sigma}_\infty$ can be approximated as the solution of the following matrix equation, called Lyapunov's equation:

$$\boldsymbol{R}_f\boldsymbol{\Sigma}_{\boldsymbol{w}}(\infty) + \boldsymbol{\Sigma}_{\boldsymbol{w}}(\infty)\boldsymbol{R}_f^H = \mu\,\boldsymbol{R}_g \tag{49}$$

where, $\boldsymbol{R}_f = \frac{d}{d\boldsymbol{w}}h(\boldsymbol{w})|_{\boldsymbol{w}=\boldsymbol{w}_*}$ and $\boldsymbol{R}_g = -\mathbb{E}\left[H(\boldsymbol{w}_*,\boldsymbol{x}(n))H(\boldsymbol{w}_*,\boldsymbol{x}(n))^H\right] = -\mathbb{E}\left[|\phi(\bar{y}(n+D))|^2\boldsymbol{x}^*(n)\boldsymbol{x}^T(n)\right]$. According to Eq.(48), we have

$$\boldsymbol{R}_f = \Big(\frac{dA(\sigma_\varepsilon^2)}{d\sigma_\varepsilon^2}\Big)[\sigma_s^2\widetilde{\boldsymbol{H}}\widetilde{\boldsymbol{H}}^H + \sigma_b^2\boldsymbol{I}_{Lw}]^T. \tag{50}$$

Let us denote $R_x = \mathbb{E}[\boldsymbol{x}(n)\boldsymbol{x}(n)^H] = \sigma_s^2\widetilde{\boldsymbol{H}}\widetilde{\boldsymbol{H}}^{\boldsymbol{H}} + \sigma_{\boldsymbol{b}}^2 I_{L_w} = \boldsymbol{U}\,\boldsymbol{\Lambda}_x\,\boldsymbol{U}^H$ the eigenvalue decomposition of $\mathbb{E}[\boldsymbol{x}(n)\boldsymbol{x}(n)^H]$, where $\boldsymbol{\Lambda}_x = \mathrm{diag}(\lambda_1, \lambda_2, ..., \lambda_{L_w})$. We can easily verify that $\boldsymbol{R}_f = \boldsymbol{U}^*\,\boldsymbol{\Lambda}_f\,\boldsymbol{U}^T$ (see Eq.(50)), where

$\boldsymbol{\Lambda}_f(i,i)_{i=1..L_w} \simeq \frac{-\lambda_i}{2N_s\sqrt{2\pi}(\sigma^2+\sigma_\varepsilon^2)^{\frac{3}{2}}}$. We can also write $\boldsymbol{R}_g \simeq \boldsymbol{U}^*\,\boldsymbol{\Lambda}_g\,\boldsymbol{U}^T$ and we detail the calculation of the diagonal elements of $\boldsymbol{\Lambda}_g$ in the Appendix (VIII-B). Thus, Eq.(49) becomes:

$$\boldsymbol{\Lambda}_f(\boldsymbol{U}^T\boldsymbol{\Sigma}_{\boldsymbol{w}}(\infty)\boldsymbol{U}^*) + (\boldsymbol{U}^T\boldsymbol{\Sigma}_{\boldsymbol{w}}(\infty)\boldsymbol{U}^*)\boldsymbol{\Lambda}_f = \mu\,\boldsymbol{\Lambda}_g \tag{51}$$

which shows that $\boldsymbol{U}^T\boldsymbol{\Sigma}_{\boldsymbol{w}}(\infty)\boldsymbol{U}^*$ is also diagonal: $\boldsymbol{\Sigma}_{\boldsymbol{w}}(\infty) = \boldsymbol{U}^*\,\boldsymbol{\Lambda}_{\boldsymbol{w}}\,\boldsymbol{U}^T$ with $\boldsymbol{\Lambda}_{\boldsymbol{w}} = \mathrm{diag}\{\lambda_{w_1}, \lambda_{w_2}, ..., \lambda_{w_{L_w}}\}$ and

$$\lambda_{w_i} \simeq \mu\frac{\Lambda_g(i,i)}{\Lambda_f(i,i)} \simeq 2\mu N_s\sqrt{2\pi}(\sigma^2+\sigma_\varepsilon^2)^{\frac{3}{2}}\mathbb{E}\left[|\phi(s(k_n))|^2\right]. \tag{52}$$

3) MSE: In this stage, we calculate the MSE. Using the classical assumption that the tap coefficient vector $\boldsymbol{w}(n)$ are independent from the input data vector $\boldsymbol{x}(n)$ [22] and that the symbols $s(n)$ are independent of the noise $b(n)$, the MSE is expanded as follows:

$$\begin{aligned}
\sigma_\varepsilon^2(\infty) &= \lim_{n\to\infty}\mathbb{E}\left[|y(n)-s(k_n)|^2\right]\\
&= \lim_{n\to\infty}\mathbb{E}\left[|\boldsymbol{w}^T(n)\boldsymbol{x}(n)-s(k_n)|^2\right]\\
&= \mathbb{E}\left[|\boldsymbol{\epsilon}(\infty)^T\boldsymbol{x}(\infty)+\boldsymbol{w}_*{}^T\boldsymbol{x}(\infty)-s(k_\infty)|^2\right]\\
&= \mathbb{E}\left[\boldsymbol{\epsilon}(\infty)^T\boldsymbol{x}(\infty)\boldsymbol{x}(\infty)^H\boldsymbol{\epsilon}(\infty)^*\right]+\boldsymbol{w}_*{}^T\mathbb{E}\left[\boldsymbol{x}(\infty)\boldsymbol{x}(\infty)^H\right]\boldsymbol{w}_*{}^*\\
&+ \sigma_s^2+2\Re\left[\mathbb{E}\left[\boldsymbol{\epsilon}(\infty)^T\right]\mathbb{E}\left[\boldsymbol{x}(\infty)\boldsymbol{x}(\infty)^H\right]\boldsymbol{w}_*{}^*-\sigma_s^2\left(\mathbb{E}\left[\boldsymbol{\epsilon}(\infty)\right]+\boldsymbol{w}_*\right)^T\widetilde{\boldsymbol{H}}\boldsymbol{e}_D\right]
\end{aligned} \tag{53}$$

As $\mathbb{E}[\boldsymbol{\epsilon}(\infty)] = 0$ and using the independence between $\boldsymbol{w}_*$ and $\boldsymbol{x}(\infty)$ again, we get:

$$\sigma_\varepsilon^2(\infty) = \mathrm{Tr}\left(\boldsymbol{\Lambda}_x\boldsymbol{\Lambda}_w\right)+\boldsymbol{w}_*{}^TR_x\boldsymbol{w}_*{}^*+\sigma_s^2-2\sigma_s^2\Re[\boldsymbol{w}_*^T\widetilde{\boldsymbol{H}}\boldsymbol{e}_D] \tag{54}$$

where $\mathrm{Tr}\left(\boldsymbol{\Lambda}_x\boldsymbol{\Lambda}_w\right)$ is the residual error of the equalizer and $\boldsymbol{w}_*{}^TR_x\boldsymbol{w}_*{}^*+\sigma_s^2-2\sigma_s^2\Re[\boldsymbol{w}_*^T\widetilde{\boldsymbol{H}}\boldsymbol{e}_D]$ is the MMSE error term [1].

VI. Simulation Results

A. Evolution law of the kernel size

The kernel size σ of the Parzen window influences the convergence speed of the algorithm and its residual error. At the beginning of convergence, it is necessary to choose a large kernel size to enable interaction of the equalized symbol with all the constellation symbols and thus ensure a fast convergence. On the contrary, when approaching the perfect equalization of transmitted symbols, a small kernel size has to be used to only allow interaction of each equalized symbol with the closest symbol in the constellation. In [12], the kernel size was adaptively controlled assuming a linear relationship between the kernel size and the decision error:

$$\sigma(n) = aG(n)+b \tag{55}$$

where, $G(n) = \alpha G(n-1)+(1-\alpha)\underbrace{min}_{k=1,\ldots,N_s}\left((|y(n)|^2-|s_k|^2)^2\right)$, α is a forgetting factor and (a,b) are empirically determined constants. In the same way, we propose to update the error $G(n)$ by $G(n) = \alpha G(n-1)+(1-\alpha)\underbrace{min}_{k=1,\ldots,N_s}\left((|y_r(n)|^2-|s_{r_k}|^2)^2+(|y_i(n)|^2-|s_{i_k}|^2)^2\right)$ when using MSQD-$\ell 2$ algorithm and by $G(n) = \alpha G(n-1)+(1-\alpha)\underbrace{min}_{k=1,\ldots,N_s}\left((|y_r(n)|-\right.$

$|s_{r_k}|)^2 + (|y_i(n)| - |s_{i_k}|)^2\big)$ when using MSQD-$\ell1$ algorithm.

As mentioned in [12], the minimum of the stochastic cost function is a scaled version of the desired constellation. Then, the original symbols $|s_k|^2$ in Eq.(22) are substituted by $|s_k^c|^2$ as follows:

$$|s_k^c|^2 = Q(\sigma)|s_k|^2 \tag{56}$$

where $Q(\sigma)$ is the compensation factor that depends on the kernel size and is obtained by ensuring that the zero-ISI solution $(y(n) = s(k_n))$ is a minimum of $\mathbb{E}\left[J(\boldsymbol{w})\right]$:

$$\mathbb{E}\left[\nabla_{\boldsymbol{w}} J(\boldsymbol{w})\right] = \frac{1}{N_s}\sum_{k=1}^{N_s} \mathbb{E}\left[K_\sigma^{'}(|s(k_n)|^2 - Q(\sigma)|s_k|^2)s(k_n)\boldsymbol{x}^*(n)\right] = 0 \tag{57}$$

For MSQD-$\ell2$ and MSQD-$\ell1$ we adopt the same approach to determine the adequate $Q(\sigma)$ for each algorithm:

$$\mathbb{E}\left[\nabla_{\boldsymbol{w}} J_{\text{MSQD-}\ell 2}(\boldsymbol{w})\right] = 0 \rightarrow Q_{\text{MSQD-}\ell 2}(\sigma)$$

$$\mathbb{E}\left[\nabla_{\boldsymbol{w}} J_{\text{MSQD-}\ell 1}(\boldsymbol{w})\right] = 0 \rightarrow Q_{\text{MSQD-}\ell 1}(\sigma).$$

Thus, the real and imaginary parts of the compensated symbols are related to the true symbols in the constellation as follows: $|s_{r_k}^c|^2 = Q(\sigma)|s_{r_k}|^2$ and $|s_{i_k}^c|^2 = Q(\sigma)|s_{i_k}|^2$. $Q(\sigma)$ is calculated numerically for each modulation. Fig.3 shows the compensation factor $Q(\sigma)$ for 16-QAM, 64-QAM and 256-QAM modulations when using the MSQD-$\ell1$ algorithm. For the MSQD-$\ell1$ and MSQD-$\ell2$, we implement the same steps as the algorithm summarized in [12], using the appropriate cost functions and Q functions.

B. Results

To compare blind equalization approaches proposed in this report, with others existing in the literature, we choosed the same channel as the one used in [12]:

$$\boldsymbol{h} = [0.2258, 0.5161, 0.6452, -0.5161]^T. \tag{58}$$

Performance of the proposed MSQD-$\ell2$ and MSQD-$\ell1$ methods are compared with those of the CMA and SQD. The latter was compared with the algorithms of CMA, Benveniste-Goursat (BG), Dual mode (DM)-CMA and its Stop-And-Go extension (SAG-DM-CMA). It was proved that the SQD method yields better performance. In the following, we show that the performance of MSQD-$\ell1$ and MSQD-$\ell2$ are better than those of SQD. Thus, the algorithms that we propose also perform better than the CMA, BG, DM-CMA and SAG-DM-CMA. For simulations, we employed an equalizer of length $L_w = 21$ initialized using the tap-centered strategy. Table (I) below summarizes the parameters which were used to draw the curves in Fig.$\{4, 5, 6\}$.

To compare the performance of the proposed algorithms in terms of convergence speed, we fixed the step size μ such as they converge with the same speed. Thus, in Fig.4, Fig.5 and Fig.6, we can clearly notice that MSQD-$\ell 2$ and MSQD-$\ell 1$ outperform the SQD and CMA algorithms in terms of residual error for 16-QAM, 64-QAM and 256-QAM modulations. On the other hand, when we oblige the algorithms to converge to the same MSE in Fig.7 and Fig.8, we notice that MSQD-$\ell 2$ and MSQD-$\ell 1$ converge faster. All these figures validate the MSQD-$\ell 1$ performance analysis that we have conducted, since the experimental curve of the MSQD-$\ell 1$ converges to the theoretical one. To study the performance of the proposed algorithms as a function of the SNR, we draw in Fig.9 the Symbol Error Rate (SER) for the CMA, SQD, MSQD-$\ell 2$ and MSQD-$\ell 1$ algorithms between SNR $= 0dB$ and SNR=20dB for a 16-QAM modulation. It is clear in this figure that the MSQD-$\ell 1$ algorithm outperforms the other algorithms in terms of the SER.

VII. Conclusion

In this report, we have proposed new criteria for kernel based blind equalization techniques that force the pdf of the real and imaginary parts of the equalizer output to match that of the true constellation real and imaginary parts by employing the Parzen window method to estimate the data pdf. Performance of the proposed methods has been compared with that of CMA and SQD. We have shown that they converge faster with a reduced residual error. The behaviour of the MSQD-$\ell 1$, most powerful proposed method, has been examined by relating the motion of the parameter estimate errors to a deterministic ODE. The analysis that we have conducted and simulation results prove that the MSQD-$\ell 1$ algorithm brings further validation of the pdf fitting approach for equalization in digital transmission.

TABLE I

PARAMETER VALUES USED FOR SIMULATIONS

16 QAM	CMA	SQD	MSQD-l2	MSQD-l1
μ	3.5×10^{-5}	10^{-4}	1.3×10^{-4}	7.7×10^{-4}
a	-	3.5	3.5	1.5
b	-	-9.5	-9.5	-1
$1-\alpha$	-	5×10^{-3}	5×10^{-3}	5×10^{-3}
E_0	-	7	7	5
64 QAM				
μ	3.3×10^{-7}	1.2×10^{-6}	9×10^{-7}	4.7×10^{-5}
a	-	3.5	3	2
b	-	-2	-18	-10
$1-\alpha$	-	10^{-3}	10^{-2}	10^{-3}
E_0	-	5	7	6.5
256 QAM				
μ	4×10^{-8}	1.5×10^{-7}	1.5×10^{-7}	7×10^{-5}
a	-	3.5	2.5	4
b	-	-4.5	-15	-1
$1-\alpha$	-	5×10^{-5}	10^{-4}	2×10^{-4}
E_0	-	7	20	7

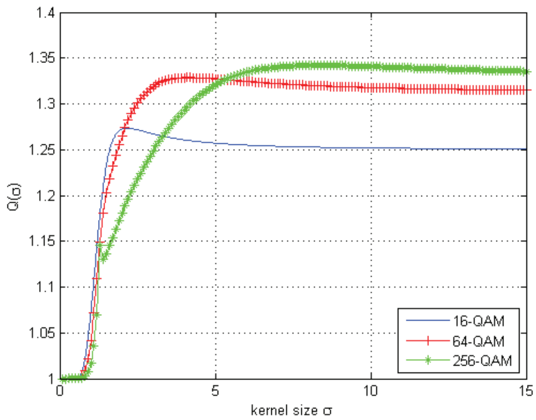

Fig. 3. Numerically obtained compensation factor $Q(\sigma)$ in the case of MSQD-l1 algorithm.

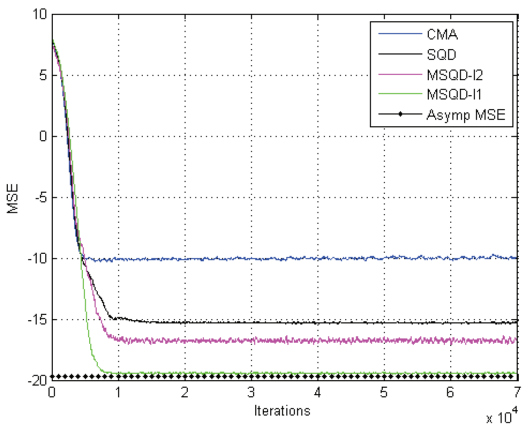

Fig. 4. MSE (dB) for 16-QAM and SNR=30 dB.

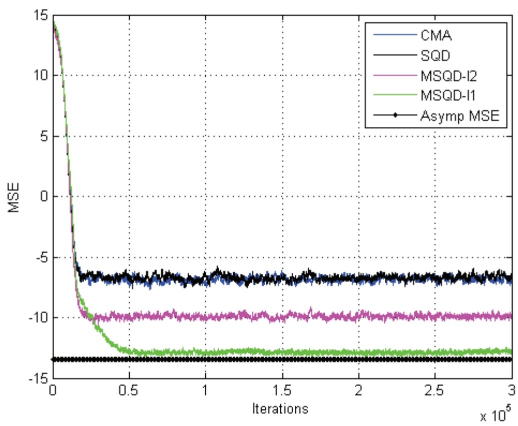

Fig. 5. MSE (dB) for 64-QAM and SNR=30 dB.

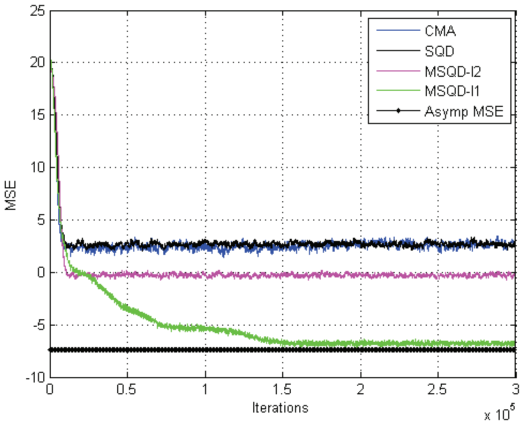

Fig. 6. MSE (dB) for 256-QAM and SNR=30 dB.

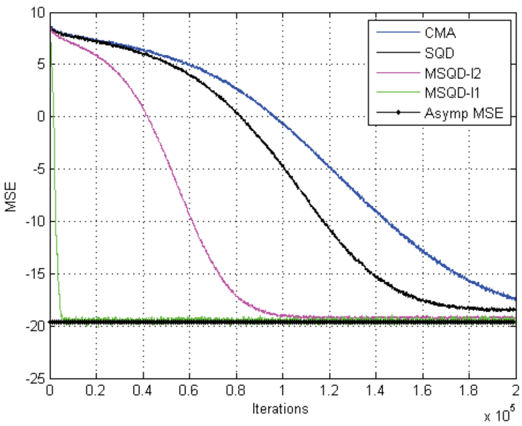

Fig. 7. MSE (dB) for 16-QAM and SNR=30 dB.

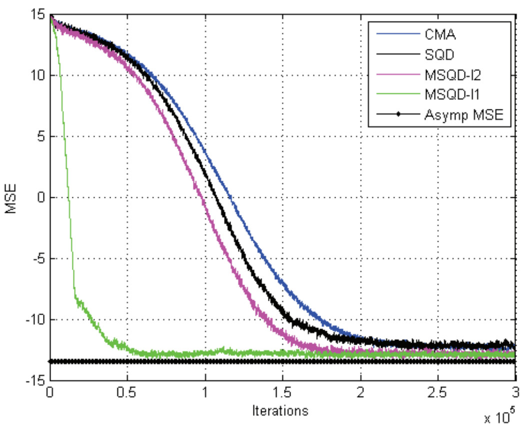

Fig. 8. MSE (dB) for 64-QAM and SNR=30 dB.

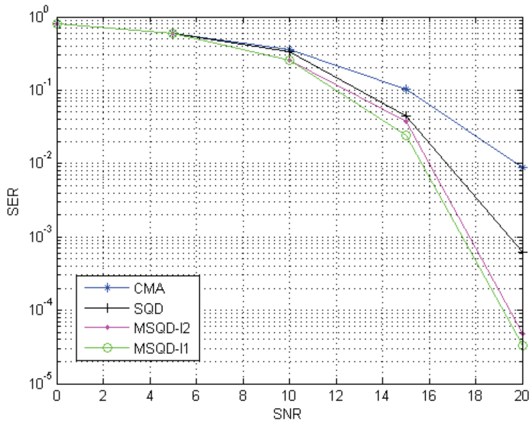

Fig. 9. SER for CMA, SQD, MSQD-l2 and MSQD-l1 algorithms.

REFERENCES

[1] A. H. Sayed, *Adaptive filters*. John Wiley & Sons, Inc., Hoboken, New Jersey, 2008.

[2] Y. Sato, "A method of self-recovering equalization for multilevel amplitude-modulation systems," *IEEE Transactions on Communications*, vol. 23, no. 6, pp. 679 – 682, June 1975.

[3] D. Godard, "Self-recovering equalization and carrier tracking in two-dimensional data communication systems," *IEEE Transactions on Communications*, vol. 28, no. 11, pp. 1867 – 1875, November 1980.

[4] J. Treichler and B. Agee, "A new approach to multipath correction of constant modulus signals," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 31, no. 2, pp. 459 – 472, April 1983.

[5] K. N. Oh and Y. O. Chin, "Modified constant modulus algorithm: blind equalization and carrier phase recovery algorithm," in *IEEE International Conference on Communications, ICC Seattle, 'Gateway to Globalization',*, vol. 1, June 1995, pp. 498 –502.

[6] ——, "New blind equalization techniques based on constant modulus algorithm," in *Global Telecommunications Conference, GLOBECOM, IEEE*, vol. 2, November 1995, pp. 865 –869.

[7] D. Jones, "A normalized constant-modulus algorithm," in *Conference Record of the Twenty-Ninth Asilomar Conference on Signals, Systems and Computers*, vol. 1, November 1995, pp. 694 –697.

[8] I. Santamaria, C. Pantaleon, L. Vielva, and J. Principe, "A fast algorithm for adaptive blind equalization using order-α renyi's entropy," *ICASSP*, 2002.

[9] J. Sala-Alvarez and G. Vazquez-Grau, "Statistical reference criteria for adaptive signal processing in digital communications," *IEEE Transactions on Signal Processing*, vol. 45, no. 1, pp. 14 –31, january 1997.

[10] I. Santamaria, C. Pantaleon, L. Vielva, and J. Principe, "Adaptive blind equalization through quadratic pdf matching," *Proceedings of the European Signal Processing Conference, Toulouse, France*, September 2002.

[11] M. Lazaro, I. Santamaria, C. Pantaleon, D. Erdogmus, K. E. Hild II, and J. C. Principe, "Blind equalization by sampled pdf fitting," *4th International Symposium on Independent Component Analysis and Blind Equalization, Nara, Japan*, September 2003.

[12] M. Lazaro, I. Santamaria, D. Erdogmus, K. Hild, C. Pantaleon, and J. Principe, "Stochastic blind equalization based on pdf fitting using parzen estimator," *IEEE Transactions on Signal Processing*, vol. 53, no. 2, pp. 696 – 704, february 2005.

[13] A. Benveniste, M. Metivier, P. Priouret, and S. Wilson, *Adaptive algorithms and stochastic approximations*. Berlin ; New York : Springer, cop, 1990.

[14] D. W. Scott, *Multivariate density estimation theory, practice, and visualization*. John Wiley & Sons, Inc., New York, 1992.

[15] E. Parzen, "On estimation of a probability density function and mode," *The Annals of Mathematical Statistics*, September 1962.

[16] I. Santamaria, C. Pantaleon, L. Vielva, and J. C. Principe, "A fast algorithm for adaptive blind equalization using order-α renyi's entropy," *ICASSP*, 2002.

[17] J. Principe, D. Xu, Q. Zhao, J. Fisher III, and C. a. Pantaleon, "Learning from examples with information theoretic criteria," *Journal of VLSI signal processing*, vol. 26, pp. 61 – 77, 2000.

[18] C. C. Cavalcante, F. Rodrigo, P. Cavalcanti, and J. C. M. Mota, "A pdf estimation-based blind criterion for adaptive equalizaion," *International Telecommunications Symposium*, 2002.

[19] J. Yang, J.-J. Werner, and G. Dumont, "The multimodulus blind equalization and its generalized algorithms," *IEEE Journal on Communications*, vol. 20, no. 5, pp. 997 –1015, June 2002.

[20] C. Laot and N. Le Josse, "A closed-form solution for the finite length constant modulus receiver," in *International Symposium on Information Theory. ISIT*, september 2005, pp. 865 –869.

[21] T. Chonavel, *Statistical Signal Processing*. Springer, 2003.

[22] L. Garth, "A dynamic convergence analysis of blind equalization algorithms," *IEEE Transactions on Communications*, vol. 49, no. 4, pp. 624 –634, April 2001.

[23] Y. Li and K. Liu, "Static and dynamic convergence behavior of adaptive blind equalizers," *IEEE Transactions on Signal Processing*, vol. 44, no. 11, pp. 2736 –2745, November 1996.

VIII. Appendix

A. Calculation of the maximum possible range for μ

Let us note $\bar{y}(n) = \boldsymbol{w}_* \boldsymbol{x}(n)$. At convergence, $y(n) - \bar{y}(n)$ is small and we can apply the taylor expansion to the function $\phi(y(n))$ (see Eq.(37)) at $\bar{y}(n)$. Then,

$$\begin{aligned} \phi(y_n) &= \phi(\bar{y}(n)) + \phi'(\bar{y}(n))(y(n) - \bar{y}(n)) \\ &= \phi(\bar{y}(n)) + \phi'(\bar{y}(n))(\widetilde{\boldsymbol{H}}\boldsymbol{s}(n) + \boldsymbol{b}(n))^T \boldsymbol{\epsilon}(n) \end{aligned} \tag{59}$$

where $\boldsymbol{\epsilon}(n) = \boldsymbol{w}(n) - \boldsymbol{w}_*$ and $\widetilde{\boldsymbol{H}}\boldsymbol{s}(n) + \boldsymbol{b}(n) = \boldsymbol{x}(n)$.

In addition, we have

$$\boldsymbol{w}(n+1) = \boldsymbol{w}(n) - \mu\, \phi(y(n))\, \boldsymbol{x}(n)^*. \tag{60}$$

Thus, substracting $\boldsymbol{w}_*$ in both sides of Eq.(60) and using Eq.(59), we find that

$$\boldsymbol{\epsilon}(n+1) = \boldsymbol{\epsilon}(n) - \mu\big(\phi(\bar{y}(n))\boldsymbol{x}(n)^* + \phi'(\bar{y}(n))(\widetilde{\boldsymbol{H}}\boldsymbol{s}(n) + \boldsymbol{b}(n))^T\boldsymbol{\epsilon}(n)(\widetilde{\boldsymbol{H}}^*\boldsymbol{s}^*(n) + \boldsymbol{b}^*(n))\big) \tag{61}$$

Taking the expectation on the both side of Eq.(61) and using the independence between $\bar{y}(n)$ and $\boldsymbol{\epsilon}(n)$, as it was assumed in [22], we get

$$\begin{aligned} \mathbb{E}\left[\boldsymbol{\epsilon}(n+1)\right] &= \mathbb{E}\left[\boldsymbol{\epsilon}(n)\right] - \mu\left(\mathbb{E}\left[\phi(\bar{y}(n))\boldsymbol{x}^*(n)\right]\right. \\ &+ \left.\mathbb{E}[(\widetilde{\boldsymbol{H}}^*\boldsymbol{s}^*(n) + \boldsymbol{b}^*(n))\phi'(\widetilde{y}(n))(\widetilde{\boldsymbol{H}}\boldsymbol{s}(n) + \boldsymbol{b}(n))^T]\mathbb{E}\left[\boldsymbol{\epsilon} n\right]\right) \end{aligned} \tag{62}$$

In [23], the authors proved that $E\left[\phi(\bar{y}(n))\boldsymbol{x}^*(n)\right] = 0$ when the cost function approaches one of its minima. Thus, Eq.(62) can be simplified to

$$\mathbb{E}\left[\boldsymbol{\epsilon}(n+1)\right] = \left(\boldsymbol{I}_{Lw} - \mu(\sigma_s^2\widetilde{\boldsymbol{H}}^*\widetilde{\boldsymbol{F}}\widetilde{\boldsymbol{H}}^T + \sigma_{\boldsymbol{b}}^2\mathbb{E}\left[\phi'(\bar{y}_n)\right]\boldsymbol{I}_{Lw})\right)\mathbb{E}\left[\boldsymbol{\epsilon}(n)\right] \tag{63}$$

where $\widetilde{\boldsymbol{F}} = \frac{1}{\sigma_s^2}\mathbb{E}\left[\boldsymbol{s}^*(n)\phi'(\bar{y}(n))\boldsymbol{s}(n)^T\right]$. Consequently

$$\mathbb{E}\left[\boldsymbol{\epsilon}(n+1)\right] = \left(\boldsymbol{I}_{Lw} - \mu(\sigma_s^2\widetilde{\boldsymbol{H}}^*\widetilde{\boldsymbol{F}}\widetilde{\boldsymbol{H}}^T + \sigma_{\boldsymbol{b}}^2\mathbb{E}\{\phi'(\widetilde{y}(n))\}\boldsymbol{I}_{Lw})\right)^n\mathbb{E}\left[\boldsymbol{\epsilon}(0)\right] \tag{64}$$

This yields the following condition upon the step size of the algorithm for convergence of the mean error :

$$0 < \mu < \frac{2}{\lambda_{max}} \tag{65}$$

where λ_{max} is the largest eigenvalue of $\sigma_s^2\widetilde{\boldsymbol{H}}^*\widetilde{\boldsymbol{F}}\widetilde{\boldsymbol{H}}^T + \sigma_{\boldsymbol{b}}^2\mathbb{E}\left[\phi'(\bar{y}(n))\right]\boldsymbol{I}_{Lw}$.

B. Diagonalization of $\boldsymbol{R}_g$ in the basis U^*

$$\begin{aligned} \boldsymbol{R}_g &= -\mathbb{E}\left[H(\boldsymbol{w}_*, \boldsymbol{x}(n))H(\boldsymbol{w}_*, \boldsymbol{x}(n))^H\right] \\ &= -\mathbb{E}\left[\phi(s(n_k))\boldsymbol{x}^*(n)\boldsymbol{x}^T(n)\phi(s(n_k))^*\right] \\ &= -\boldsymbol{H}^*\mathbb{E}\left[|\phi(s(n_k))|^2\boldsymbol{s}(n)\boldsymbol{s}^T(n)\right]\boldsymbol{H}^T - \mathbb{E}\left[|\phi(s(n_k))|^2\boldsymbol{b}^*(n)\boldsymbol{b}^T(n)\right] \\ &= -\boldsymbol{H}^*\boldsymbol{D}\boldsymbol{H}^T - \sigma_b^2\mathbb{E}\left[|\phi(s(n_k))|^2\right] \end{aligned} \tag{66}$$

where, $\boldsymbol{D} = \text{diag}(d_1, ..., d_1, d_2, d_1, ..., d_1)$ with $d_1 = \sigma_s^2\mathbb{E}\left[|\phi(s(n_k))|^2\right]$ and $d_2 = \mathbb{E}\left[|s(n_k)|^2|\phi(s(n_k))|^2\right]$. We have cheked numerically that $|\frac{d_2-d_1}{d_1}|^2$ is very small (around 10^{-4}). Then, we can consider that $D \simeq d_1\boldsymbol{I}_{L+L_w-1}$. Thus, we obtain the following expression of R_g:

$$\begin{aligned} R_g &\simeq -d_1 \boldsymbol{H}^* \boldsymbol{I}_{L+L_w-1} \boldsymbol{H}^T - \sigma_b^2 \mathbb{E}\left[|\phi(s(n_k))|^2\right] \\ &\simeq -\mathbb{E}\left[|\phi(s(n_k))|^2\right]\left(\sigma_s^2 \boldsymbol{H}^* \boldsymbol{H}^T + \sigma_b^2 \boldsymbol{I}_{L_w}\right) \\ &\simeq -\mathbb{E}\left[|\phi(s(n_k))|^2\right]\left(\boldsymbol{U}^* \boldsymbol{\Lambda}_x \boldsymbol{U}^T\right) \\ &\simeq \left(\boldsymbol{U}^* \boldsymbol{\Lambda}_g \boldsymbol{U}^T\right) \end{aligned} \tag{67}$$

where $\boldsymbol{\Lambda}_g(i,i) \simeq -\mathbb{E}\left[|\phi(s(n_k))|^2\right] \lambda_i$.

Campus de Brest
Technopôle Brest-Iroise
CS 83818
29238 Brest Cedex 3
France
+33 (0)2 29 00 11 11

www.telecom-bretagne.eu

Imprimé à Télécom Bretagne
Dépôt légal : Juin 2013
ISSN : 1255-2275